Continuous Flash

Hugues Hoppe   Kentaro Toyama

October 1, 2003

Technical Report
MSR-TR-2003-63

Microsoft Research
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

Abstract

To take a picture of scene, we acquire multiple images under different levels of flash intensity (possibly just "flash on" and "flash off"), and let the user subsequently adjust the flash level continuously by interpolating among the stored images. Camera motion is corrected using image registration techniques robust to intensity differences. Flash adjustment occurs either immediately on the imaging device or subsequently on a PC, or the multiple acquired images are stored indefinitely in compressed form. The user adjusts flash intensity using a simple slider control, either uniformly over the whole image or regionally by segmenting the image or identifying image layers.

Introduction

When taking pictures with either a digital or an analog camera, the use of a flash increases the illumination of the scene. This added illumination is essential when the environment is too dark. Moreover, it provides "fill-in" illumination of subjects against brighter backgrounds, and reduces the intense contrast of shadows.

However, obtaining the correct amount of flash often proves difficult, since the flash intensity can be either too dark or too bright and cannot be controlled by the photographer *during* shooting in any meaningful manner. While the flash is sometimes insufficient to illuminate the scene, at other times it overexposes the subject. The problem is particularly challenging when the scene comprises both near and far objects, or objects with widely different reflection characteristics.

It's important to note that because the effect of flash on an object in the scene is dependent on the object's distance to the camera and flash apparatus, its bi-directional reflectance function, and the clarity of the atmosphere, changing the degree of flash has an effect that is qualitatively different from optical exposure settings, post-hoc contrast enhancement, and other physical or digital illumination adjustments that are commonly made in photography.

We describe a *continuous flash* approach, which facilitates user-selectable illumination of an existing picture ranging from ambient light illumination to full artificial flash illumination and every gradation in between (as well as some physically unrealizable "negative" flash effects). The end result is a *flash-adjusted* image.

Our basic approach

We acquire and store two images with a digital camera:

– one taken without flash, and

– one taken with flash.

Ideally, these images are acquired at the same time (or within hundredths of a second) using the same set of view and exposure parameters.

Subsequently, we let the user interpolate (and extrapolate) between these two stored images, to adjust the amount of flash illumination to an optimum level.

Notation

Let I_0 denote the image taken without flash, and I_1 denote the image taken with flash.

We define a continuous family of images:

$I(\alpha) = (1-\alpha) I_0 + (\alpha) I_1$,

where $\alpha \in R$ represents the quantity of flash.

Thus,

$I(0) = I_0$ is the original image taken without flash,

$I(1) = I_1$ is the original image taken with flash,

$I(\alpha)$ for $0<\alpha<1$ is an interpolated image where flash intensity is adjusted between the two acquired images,

$I(\alpha)$ for $\alpha>1$ is an extrapolated image that has more flash than taken with the original image with flash, and

$I(\alpha)$ for $\alpha<0$ is an extrapolated image that has less flash than possible (i.e. subtracts flash reflectance from an image already taken without flash).

Example

Acquired pair of images:

image $I_0=I(0.0)$ without flash

image $I_1=I(1.0)$ with flash

Subsequently interpolated (and extrapolated) images:

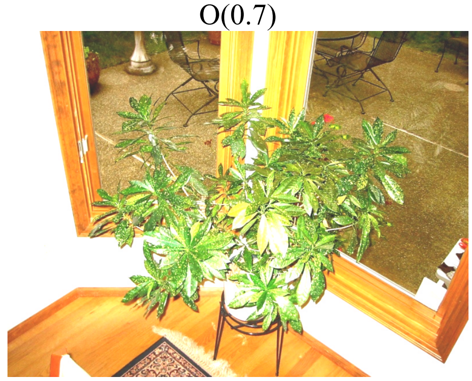
I(-1.0)

I(0.3)

I(0.5)

O(0.7)

I(1.4)



I(2.0)

Related work

Debevec [1997] acquires several images of a scene under different exposure settings, to construct a high dynamic range (HDR) image of the scene.

Toyama et al [2003] acquire multiple images of a scene under different aperture and focal-length settings, and let the user subsequently interpolate between the settings. However, they do not consider the use of flash.

Cohen et al [2003] acquire series of still images using a tripod-mounted camera. They apply a variety of filters to the resulting "image stack", and combine the results using 2D brush painting operations. Applications include merging of video stills, high dynamic range imagery, and the creation of group pictures incorporating multiple imperfect stills.

Discussion

Multiple flash intensities

The set of acquired images can include several images with the flash at different intensity settings, in addition to the image without flash. Having more images may improve the quality of subsequent interpolation.

Exposure settings

The multiple source images should be captured with identical exposure settings. Or, at least it is preferable to know the "F-stop" difference between the two exposures, so that the image values can be easily brought into a common luminance coordinate system prior to interpolation.

High dynamic range

Optionally, for each setting of the flash, multiple images at different exposure settings are acquired to construct an HDR image.

Image registration

The two or more images should be acquired in a short time interval, so that there is little motion between them. If the images are taken a few hundredths of a second apart, the motion is likely tolerable.

Otherwise, a tripod can be used to hold the camera steady between the two pictures. But even if the camera is held perfectly steady, objects in the scene may themselves be in motion, so this must still be rectified.

The camera or local scene motion can be compensated for by a process of *image registration*. This problem is well studied in computer vision. The one big difference in our context is that the images generally have different intensity values, since each one is taken with a different flash intensity. Thus, the registration algorithm should be designed to be robust to intensity differences. For instance, one can apply a Gaussian or Laplacian filter to both images to locate high-derivative regions of the image, and align these derivative images together. For additional robustness, the alignment cost metric could examine just the signs of the derivatives and ignore their magnitudes. For recent work on image registration in the presence of exposure differences, see [Kang et al 2003].

Interpolation function

Simple linear interpolation in RGB color space already provides satisfactory results. One can also consider interpolating using other color models, such as HSV (hue, saturation, value) or HLS (hue, lightness, saturation).

With images acquired at multiple flash intensities, let the images be denoted I_0, I_1, …, I_n, where the flash intensity in image I_i is f_i (some continuous value). Thus, for the common two-image case, one simply has images I_0, I_1 with f_0=0 and f_1=1 (zero flash and full flash). One example of an image set with greater than two images is a set of 3 images: I_0, I_1, I_2 with f_0=0, f_1=0.5, f_2=1. During the interpolation, each pixel is treated independently. That is, for a given pixel, it has a different color value in each of the images. These values are denoted c_0, c_1, … c_n. The result is thus a simple data fitting problem (e.g., curve fitting or regression). A function c(f) that fits the two-dimensional points (2D) (f_0,c_0), (f_1,c_1), … (f_n,c_n) is sought. One example of such a function is a piecewise linear curve that just connects the points. This piecewise linear curve then corresponds to linear interpolation between successive samples. Another example of a suitable curve for interpolation is a spline curve (a polynomial function that is fit to the 2D points). There are approximating spline curves (that do not pass through the points, e.g. cubic B-spline), and interpolating spline curves (that do pass through the points, e.g. Catmull-Rom spline).

Flash adjustment scenarios

Flash adjustment can occur within the digital camera itself, using appropriate software or firmware.

Or, it can occur later when the images are uploaded to a PC, for instance when they are entered into a photo album.

Or, the multiple images can be all preserved, to allow flash adjustment at any time in the future. Because the multiple images are so much alike, especially after alignment, compressing them together results in a representation that does not require much more space than storing just one image alone. For example, a mean image can be compressed and then differences images can be compressed, resulting in a very low storage cost.

User interface

The simplest interface is to provide the user with a slider that adjusts the flash intensity uniformly over the whole image. Here is an example:

Automatic adjustment

The optimal flash intensity could be selected automatically, incorporating the output of a machine-learning algorithm that has been trained on sets of images which serve as examples of “good” and “bad” levels of flash.

Region-based adjustment

In many cases, it is useful for the user to adjust the flash intensity differently in various regions of the image. For instance, the foreground object may have been over-exposed in the flash image, while the background was underexposed in the no-flash image. To this end, image processing techniques can segment the image into foreground and background regions, or even segment the image into layers. The user then adjusts the flash intensity independently on the resulting segments or layers.

A different approach is to combine the multiple source images using a paintbrush interface as in [Cohen et al 2003].

References

COHEN, M., COLBURN, R., AND DRUCKER, S. 2003. Image stacks. Microsoft Research MSR-TR-2003-40, July, 2003.

DEBEVEC, P. 1997. Recovering High Dynamic Range Radiance Maps from Photographs. *ACM SIGGRAPH 1997*.

KANG, S.B., UYTTENDAELE, M., WINDER, S., AND SZELISKI, R. 2003. High dynamic range video. *ACM SIGGRAPH 2003*, 319-325.

TOYAMA, K., AND SCHOELKOPF, B. 2001. Interactive Images. Microsoft Research MSR-TR-2003-64, October 1, 2003.